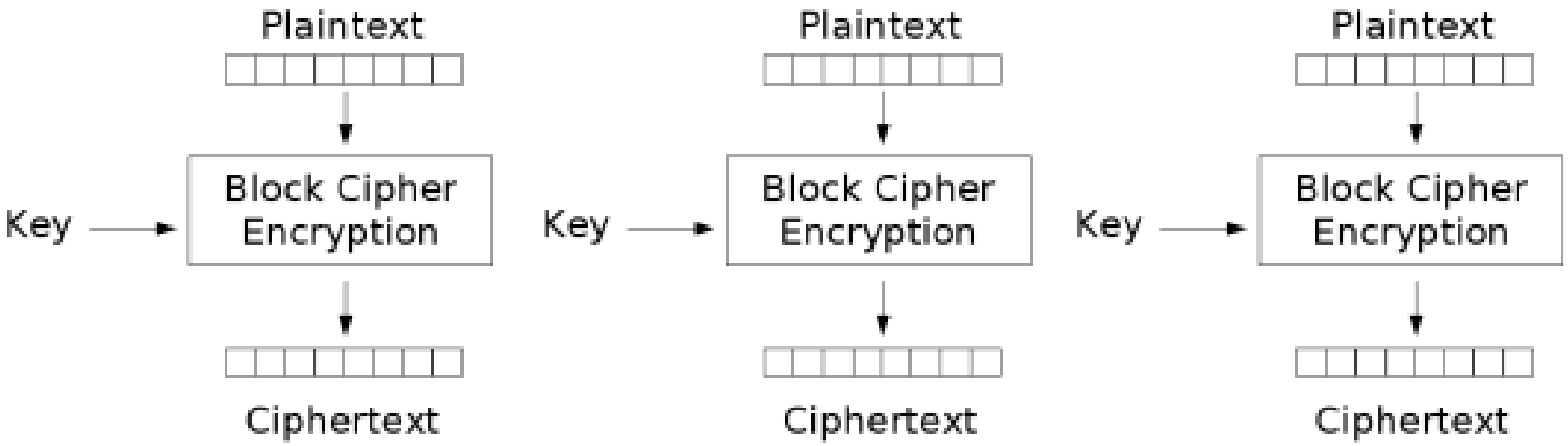
15-Minute Intro to Cryptography

Symmetric encryption

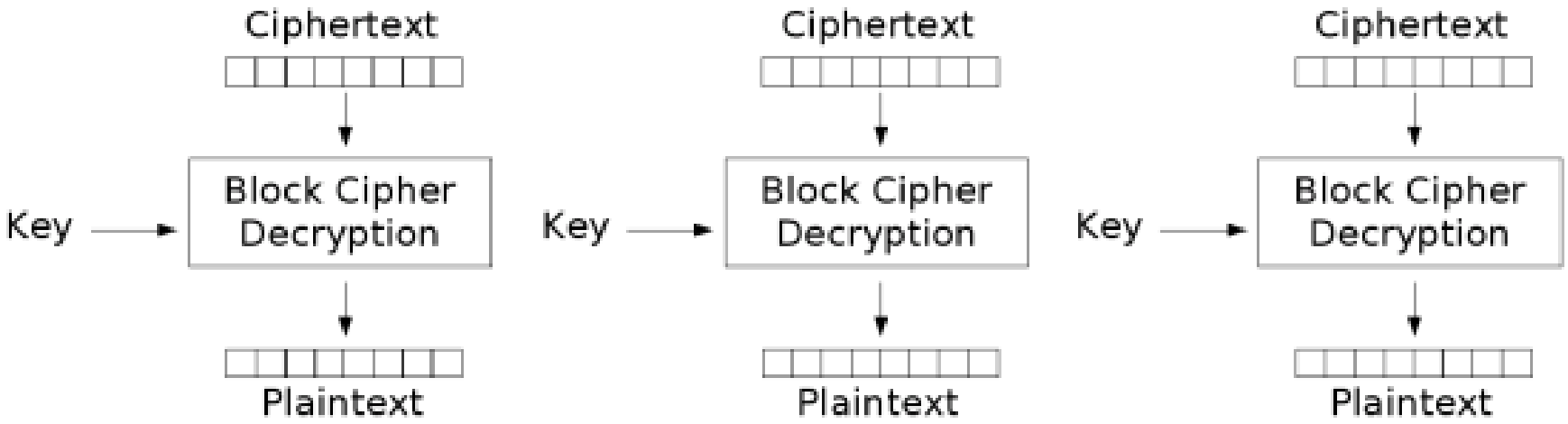
- Enc(Data, Key) -> Ciphertext
- Dec(Ciphertext, Key) -> Data
- Algorithm is public
- Key is secret
  - Usually 128..512 bits
- Operates on fixed-size blocks
  - Usually 56 or 128 bits
- Summary: reduces big secret (data) to small secret (key)

Encrypting large amounts of Data



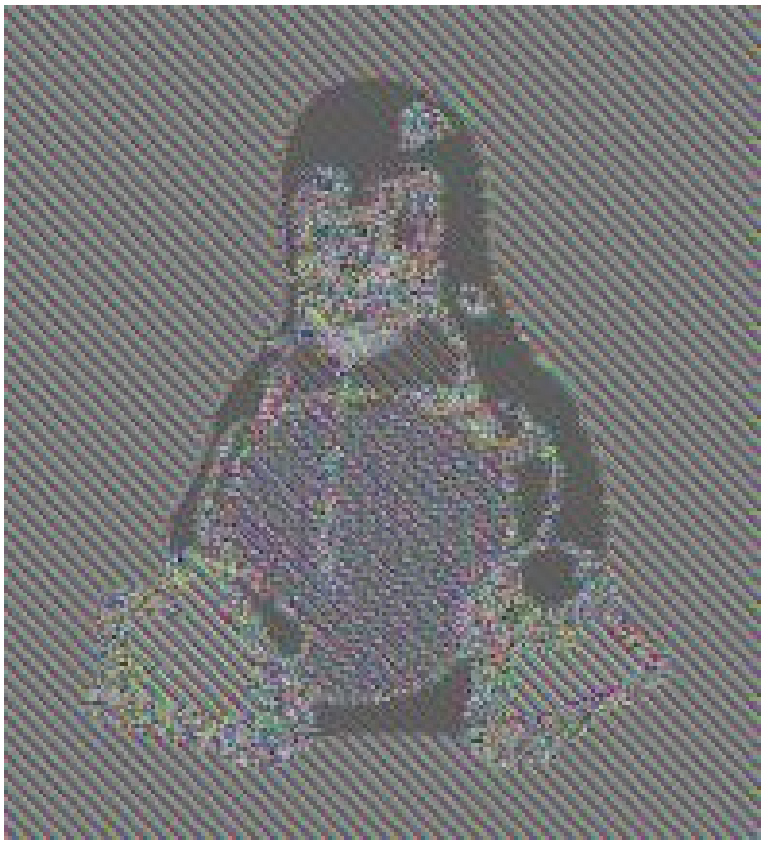
Electronic Codebook (ECB) mode encryption

Encrypting large amounts of data (2)

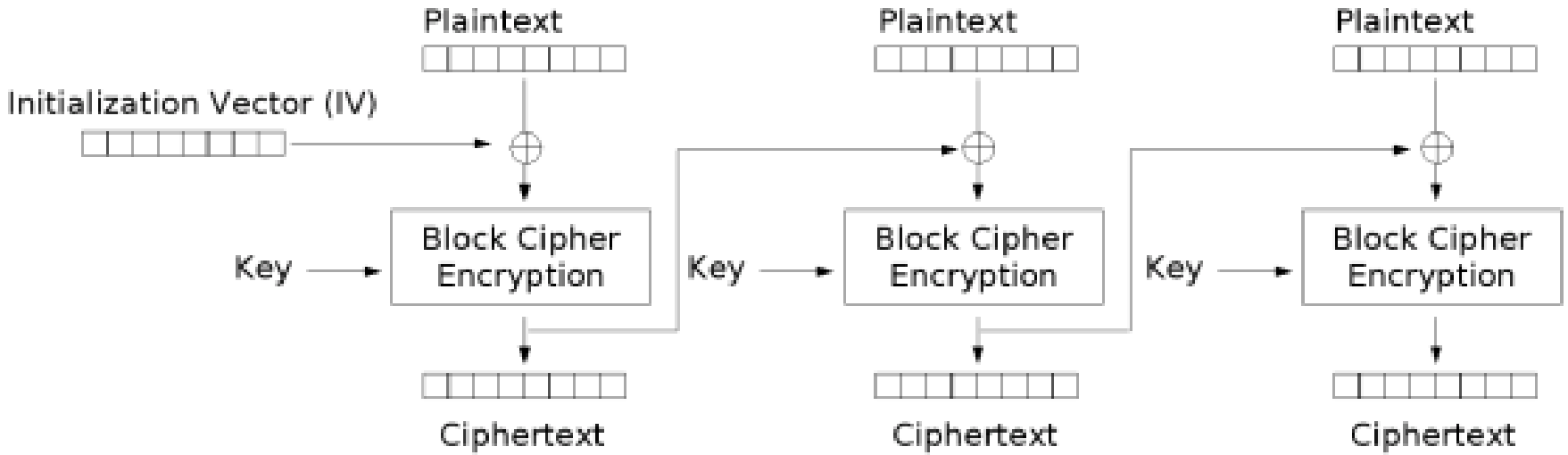


Electronic Codebook (ECB) mode decryption

Encrypting large amounts of data (3)

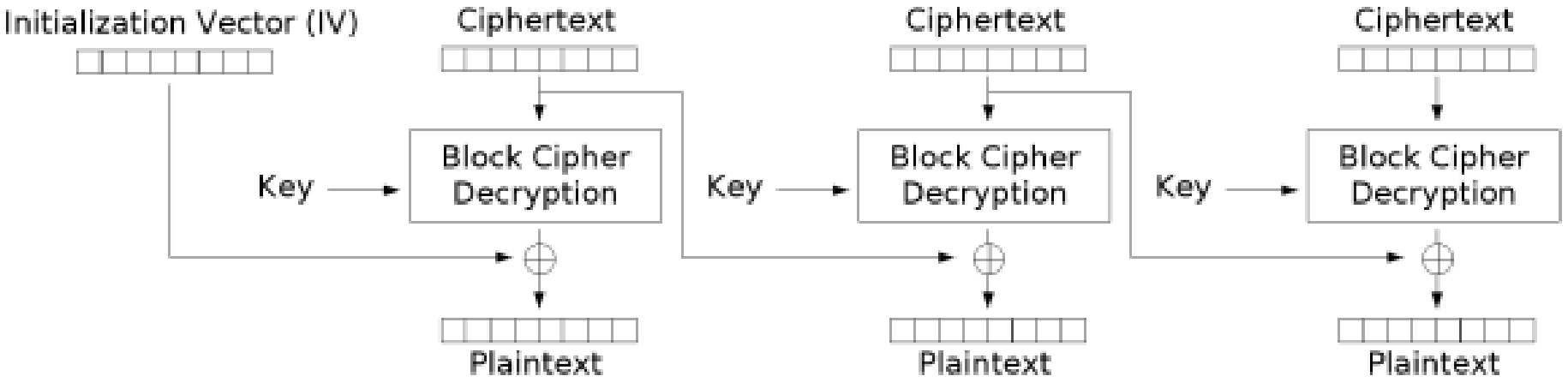


Encrypting large amounts of data (4)



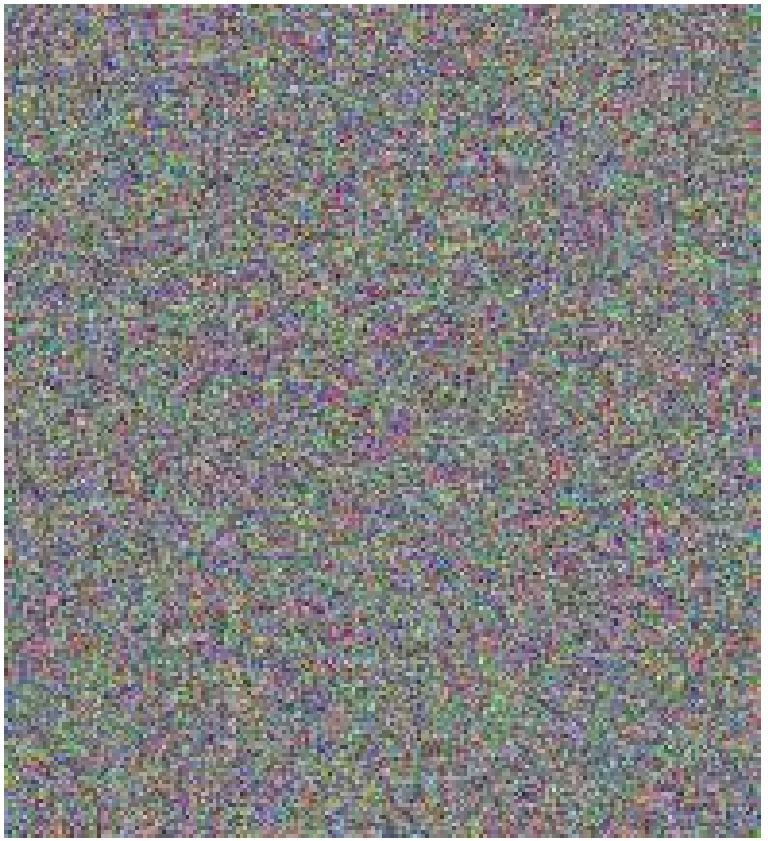
Cipher Block Chaining (CBC) mode encryption

Encrypting large amounts of data (5)



Cipher Block Chaining (CBC) mode decryption

Encrypting large amounts of data (6)



Asymmetric encryption

- Key consists of two parts: Pub and Priv
  - Public key for encryption, private key for decryption
  - (Practically) impossible to derive Priv from Pub
- Enc(Data, Pub) -> Ciphertext
- Dec(Ciphertext, Priv) -> Data
- The trick is to use correct public key for encryption
- Reduces big secret to small authentic data

Digital signature

- Also asymmetric operation with private and public key
- Sign(Data, Priv) -> Signature
- Verify(Signature, Data, Pub) -> Yes/No
- Reduces big authentic data to small authentic data

Hashing

- Hash(Data) -> Hashed
- Input can be any length
- Output is fixed length (e.g. 120 or 256 bits)
- (Practically) impossible to find Data1 and Data2 that give same output
- (Practically) impossible to find Data that outputs given Hash
- Reduces big authentic data to small authentic data

Message Authentication Code

- MAC(Data, Key) -> Code
- VerifyMac(Data, Code, Key) -> Yes/No
- Keyed hash function
- Symmetric key
- Used for quick integrity check in on-line protocols

Conclusion

- Building blocks for building more complicated protocols
  - Symmetric encryption
  - Asymmetric encryption
  - Digital signatures
  - Hashing
  - MAC